PARTS:
ARDUINO
SOLID STATE RELAY (3.95) OR POWER SWITCH TAIL (19.95)
14 GAUGE WIRE
LIGHTING (LED LAMP)
USRF SENSOR(S)

INPUT DATA (in range)
60 ≤ USRF SENSOR ≤ 100 = PIN 1
120 ≤ USRF SENSOR ≤ 160 = PIN 2
180 ≤ USRF SENSOR ≤ 220 = PIN 3
240 ≤ USRF SENSOR ≤ 280 = PIN 4
300 ≤ USRF SENSOR ≤ 341 = PIN 5

Basic idea for script:
Range = Object within distances (above)
Y = Time within Range
If USRF SENSOR = Range;
Turn on Pin;
Else (USRF SENSOR)=Range. For Y, amount of time
Turn on Pin for Y, amount of time

PREDESIGN:
ABOVE:
INSTALLATION DIAGRAM:
This project’s goal was to create responsive lighting that has a memory, or a residual affect on a space. Generally, lights are responsive to the amount of time a user occupies its lit subject (this case, wall art).

This uses a series of overhead lights, 2 analog inputs, 120VAC lighting and relays to control ON/OFF of the live wire, and the Arduino to supply a script to control the relays via DC from digital pin outputs.
ARRANGEMENT:

ABOVE:

ULTRA SONIC RANGE FINDERS:
The initial idea was to use 2 range finders to allow for at least two people to be detected at once. The arrangement of the sensors went through the iterations of:
• Hanging in the space and pointing at each other
• Centered in a space, facing away from each other
• Mounted at the bottom of the wall, pointing out and towards each other

The last iteration was chosen for its ability to conceal the sensor or, at least, keep it out of sight.

These sensors are plugged into analog pins 0 and 1 and give a serial data feed that is averaged in the script. These sensors produce a lot of noise in the data and averaging was one of the most crucial steps to using these.
KEY COMPONENTS:

ABOVE:
- ARDUINO UNO
- 2 MAXBOTIX RANGE FINDERS

TOP RIGHT:
- 1K RESISTORS
- NPS222A TRANSISTORS

MIDDLE RIGHT:
- CROUZET OAC5 SOLID STATE RELAYS (120VAC/5VDC SWITCH)

BOTTOM RIGHT:
- 4 3WIRE EXTENSION CORDS (FOR RELAYS)

NOT PICTURED:
- 40 WATT BULBS (EACH SSR CAN HANDLE THREE OF THESE MAX)
- 1 5OUTLET STRIP
- 1 9V WALL WART
- BULB SOCKET/REFLECTOR
- LOTS OF WIRE
- 4LEDS (FOR DEBUGGING)
- BREADBOARD
MAINS (120VAC):

RIGHT

SWITCHING ON/OFF:
The SSR is spliced into the extension cord. The positive (black) wire is cut and soldered to the in/out of the relay leads. Positive and negative DC wires are also soldered onto the relay to connect to the Arduino.

Before anything was plugged into the wall, I verified the wiring of the relay by connecting a multimeter (in continuous mode) to see that the Arduino is switching the relay on and off. Then I waited a while, checked it again, crossed my fingers, then plugged it into the wall outlet.

ABOVE
After not getting anything to work, I realized the arduino was not putting out enough voltage from its digital pins, even though the SSR’s could be triggered by 3V they seemed to want at least 5V (see Multimeter’s read of 4.95). The relay was then connected via breadboard to the Arduino using a transistor and resistor: the transistor to boost the digital pin voltage to the relay (the Arduino would not produce enough (5V)) and a resistor to block any back voltage from the relay—to protect the Arduino. Initially, the relay was triggered by any movement from the USRF sensor. PIR sensors were also tried however they produced extremely variable results and their semi-spherical range was too wide for the intended use.
LIGHTS:
BULBS, WATTS, TYPE:
One use for the project is to limit extraneous energy use from lighting that is constantly turned on. It was my initial thought to use LED spots for the project however I was unable to find a bulb with more than 500 Lumens for under 20$. With this extremely high price point, I chose lower wattage (40W) incandescent bulbs. CFL bulbs were not considered because they cannot handle excessive on/off abuse.

For simplicity, each bulb has its own relay/cord/outlet/DC control.
SCRIPT:
ENDLESS DEBUGGING:
The script uses a series of IF statements to define states in which lights are triggered on/off. Within this is a counter that keeps track of how many times in each state is consecutively activated.

The initial state of each light is defined by the analog read of either the right or left sensor.

ie. LeftSensor >100 will turn on a corresponding light that is placed within the range of 100 units.

The counting is to use in defining secondary states of the lights. The amount of time the light stays on once the user leaves corresponds to the amount of time they spent activating the light.
WIRING:
Everything wired:
RANGE FINDING:
ABOVE

Lots of Tuning, Lots:
From the two range finders placed nearly on the floor (arrow in drawing showing the Left Sensor) an elliptical sensing area around the installation. The tuning involved calibrating the 8 states (4 stages per side) to fit within a range specified by the distance between lights:

Single States:
- L1=ON + L2=OFF
- L1=ON + L2=ON,
- L1=OFF + L2=ON,
- L1=OFF+ L2=OFF
- R1=ON + R2=OFF
- R1=ON + R2=ON,
- R1=OFF + R2=ON,
- R1=OFF+ R2=OFF

(All states can be paired with another if two users are present)

Each state corresponds to a distance from the left and right side of the installation; they meet in the middle where a ninth state of L2=ON + R1=ON can be activated by one person.
INSTALLATION:
In the end, the four lights were wall-mounted and spaced at equal intervals with a photograph hung in front of each light.

A second sheet with a letter was placed behind the photo, spelling CALM when illuminated, enticing users to activate all lights to understand the residual effect of their presence.